Project Title: Swish : Private Social Sharing Platform for Campus Communities
Project Code: SOC-WEB-2025-094

Overview
Swish is a full-stack web application designed to bring the energy of social media into a secure, campus-exclusive environment. The platform allows students and faculty to share photos, updates, and achievements within their academic network. It mirrors the fast, visual, and engaging experience of modern social apps while ensuring privacy and moderation suitable for institutional use.

The goal is to create a vibrant digital space that celebrates campus life - where every post, comment, and like connects people within the same community.

Project Goals
• Build a MERN-based social networking platform limited to verified campus users.
• Enable users to post images, captions, and short updates visible to their network.
• Implement engagement features such as likes, comments, and follow system.
• Provide an admin dashboard for content monitoring and moderation.
• Deliver a modern, mobile-responsive interface with a clean, minimal aesthetic.

Technical Scope
Develop the project using **MongoDB, Express, React, and Node.js**.
Implement JWT-based authentication and role-based access control for Students, Faculty, and Admins.
Use secure file handling (e.g., Multer or Cloudinary) for image uploads.

Core Features

1. **User Profiles:** Registration and login with profile photo, bio, and basic info.
2. **Post Feed:** Create, view, and interact with image-based posts.
3. **Like & Comment System:** Real-time engagement features integrated with the post feed.
4. **Explore Section:** View trending posts or profiles across the campus.
5. **Admin Controls:** Moderate content, handle reports, and manage user activity.
6. **Notifications (Optional):** Basic alerts for likes, comments, and follows.

Expected Deliverables
• Functional prototype of Swish demonstrating posting, feed, and interaction features.

• RESTful backend APIs for authentication, posts, and comments.
• React-based frontend with responsive design and fluid navigation.
• Documentation covering data models, API endpoints, and architecture.
• Demo presentation highlighting the user experience and technology stack.

Expected Outcomes

• Deliver a campus-exclusive social media platform encouraging creative expression and engagement.
• Strengthen understanding of authentication, file uploads, and frontend-backend data flow.
• Improve skills in scalable web architecture and UI design.
• Provide real-world experience in building social-network-style applications.
• Create a deployable, visually appealing product suitable for institutional environments.

Approach Guidelines

• Define collections for Users, Posts, and Comments before backend setup.
• Implement authentication and post creation as core modules before adding engagement features.
• Prioritize responsive UI and intuitive navigation for easy interaction.
• Conduct basic testing for performance, media handling, and security.
• Prepare a final demo showing both user and admin perspectives.